

GIT

1 Git is version control system

--> untuk membuat beberapa versi dari project

--> untuk bekerja dalam system

Intro commands

1 `git --version`

--> untuk melihat versi git

2 `git init`

--> untuk memulai git di repo tempat command line berada

Commit commands

1 `git status`

--> memberi tahu status codingan kita (udah di track belum)

2 `git add .` atau `git add nama_file`

--> untuk track semua atau sebagian file

3 `git commit -m "apa komentarnya"`

--> untuk commit code-code yang sudah di track sekalian kasih komentar

4 `git log`

5 `git log --oneline`

--> untuk melihat id commit dan komentarnya

6 `git dif`

--> untuk melihat perbedaan codingan antara yang sudah di track sama yang belum di track (bisa lihat penambahan line atau pengurangan line)

Undo and Redo

1 `git checkout nama_file`

--> undo ke posisi terakhir kita commit

2 `git checkout id_commit -- nama_file`

--> undo / redo ke commit dengan id tersebut

Cara pindah ke log lain sekalian menghapus log-log diatasnya

3 `git reset --mixed id_commit`

--> hapus tapi working directory yang kita buka, codingannya nggk berubah

4 `git reset --hard id_commit`

--> hapus dan working directory yang kita buka, codingannya kembali ke saat kita commit dengan id_commit_ini

Percabangan

1 `git branch`

--> Untuk melihat branch-branch yang ada

2 `git branch nama_brach_baru`

--> membuat brach baru

3 `git checkout -b nama_branch`

--> pindah ke branch yang dibuat dan membuat brach tersebut

4 `git checkout nama_branch`

--> pindah ke brach tersebut

5 `git branch -D nama_branch`

--> menghapus branch

6 `git branch -m nama_brach_lama nama_brach_baru`

--> mengubah nama branch

Memasukkan brach cabang ke branch Master

1 `git merge nama_cabang`

Conflict

--> Jika dalam satu file, kita melakukan perubahan pada branch master dan kita juga melakukan perubahan pada brach cabang yang ternyata coflict, maka akan terjadi git conflict

--> contoh :

di master:

<h1>Nama</h1>

<h2>Anto</h2>

di cabang:

<h1>Nama</h1>

<h2>Janjang</h2>

Jika kita merge cabang, akan ada conflict pada line 2 menjadi:

<h1>Nama</h1>

<<<<<<HEAD

<h2>Anto</h2>

=====

<h2>Janjang</h2>

>>>>>>cabang

--> Tinggal merubah kode ini manual, lalu lakukan merge lagi, maka conflict akan teratasi

GITHUB

A Repository dan Clone

1. cara clone:

--> `git clone url nama_baru`

B PUSH and PULL

1. cara push:

- `git add .`
- `git commit -m "keterangan"`
- `git remote add origin (perlu nggk si(?))`
`git push origin -u nama_brach(red:biasanya master)`
- `git push origin nama_brach(red:biasanya master)`
--> origin itu remote repository kita

2. cara pull:

--> `git pull`
--> buat ngambil yang terbaru dari github

C BRANCH and MERGE

1. membuat branch baru:

- `git checkout -b nama_branch`

2. cara merge dari brach ke master:

- klik compare samping branch
- create pull request
- create pull request
- Merge pull request
- Delete brach

D FORK

--> Ngambil repo orang ke repo kita

E .GITIGNORE

--> file yang isinya nama2 file yang kita nggk mw keliatan di repo kita(misal nama passsword db)

cara :

- buat file `.gitignore`
- masukin nama file yang mw disembunyiin

=====

`pages.github.com`

--> tempat hosting gratis buatan github

--> kekurangan : nggk bisa dinamis, cuma bisa statis

--> cara :

- buat repo dengan nama `nama_akun.github.io`
- masukin file2 htmlnya